

PROGRAMACIÓN ALGORITMOS Y ESTRUCTURAS DE DATOS

PRACTICA N º2

Programación Estructurada en Lenguaje C

- 1) Familiarícese con el ambiente de trabajo (IDE) Code::Blocks y el proceso de codificación, compilación y ejecución de un programa:
 - a) Cree un proyecto nuevo, prestando atención a la carpeta donde ubica el proyecto. Tras crearlo, busque la carpeta del proyecto e identifique
 - i. El archivo de código C con la plantilla creada por defecto
 - ii. El archivo del proyecto de Code::Blocks
 - b) Compile el programa que el entorno crea por defecto y verifique que no haya errores ni advertencias. Verifique que en la carpeta de proyecto se hayan creado nuevas carpetas. ¿Dónde está el ejecutable?
 - c) Ejecute el programa desde el entorno, y luego desde el ejecutable que encontró en la carpeta. ¿Qué diferencias observa?

Nota: Puede utilizar el apunte “Tutorial Code::Blocks” que encontrará en la página de la cátedra para ayudarse.

- 2) Se tiene el siguiente fragmento de código:

```
char a = 120; /* Se declara la variable "a" y se inicializa */
printf("a: %d", a); /* Se imprime en pantalla el valor de a */
```

 - a) Prográmelo y ejecútelo en el IDE para corroborar que hace lo que espera.
 - b) Cambie el valor de inicialización de **a** por 150. ¿Qué sucede?
 - c) Busque cuál es el máximo valor positivo que puede almacenar en **a**. Puede utilizar este programa para “ir probando” o puede consultar la teoría
 - d) Cambie el tipo de **a**, en lugar de **char** pruebe con **unsigned char** y repita los casos anteriores para 120, 150 y 260. Comente lo observado.
- 3) Indique con que tipo(s) de dato(s) (**char**, **short**, **int**, **float**, **double**, etc.) declararía variables para almacenar cada uno de los siguientes valores:
 - a) 3.1416
 - b) 5000
 - c) '3'
 - d) 3
- 4) Realice el código que
 - a) Declare las variables del punto anterior
 - b) Les asigne los valores correspondientes
 - c) Imprima en pantalla dichos valores en el formato adecuado
 - d) Repita el ejercicio, pero en el inciso b) permita que el usuario ingrese el valor de cada variable por teclado

- 5) Realice el programa que luego de ejecutarse se verá como sigue:

```

***   Calculadora de panqueques   ***
*** Encante a sus invitados con unos ***
*** buenos panqueques             ***

¿Cuántos invitados tiene? (1-100)
Ingrese el número: 6

¿Cuantos panqueques comerá cada invitado? (1-10)
Ingrese el número: 3
La receta para 6 invitados y 3 panqueques por invitado es:

      Ingrediente      cantidad      unidad
      -----
      Huevos           3,0           unidades
      Leche             500,0        mililitros
      Harina            250,0        gramos
      Sal               1,0          cucharaditas
      Vainilla          1,0          chorritos

Que disfrute los panqueques!

```

Nota: Para imprimir la tabla puede usar la secuencia de escape \t en el printf

- 6) Analice qué se imprimirá en pantalla al ejecutar los siguientes fragmentos de código, luego ejecútelos y compruebe sus predicciones. Nota: Si copia y pega el código desde el PDF, esté atento a posibles errores de copiado que pueden impedir la compilación.

a) `int metros;`
`float kilometros;`
`printf("Ingrese una distancia en metros para convertirla: ");`
`scanf("%d",&metros);`
`kilometros = metros/1000;`
`printf("Resultado: %f",kilometros);`

b) `int a;`
`a=0;`
`printf("%d\n",a++);`
`printf("%d\n",++a);`
`a++;`
`printf("%d\n",++a);`

c) `char x;`
`x = 'h';`
`printf("%c %d\n",x,x);`
`x = 105;`
`printf("%c %d",x,x);`

- 7) Encuentre los errores en los códigos disponibles en la página de la cátedra (en la carpeta Descargas/CodigosP2) y corrijalos para asegurar que los programas compilen y se ejecuten apropiadamente.
- 8) Codifique en lenguaje C los algoritmos planteados en la práctica N° 1, desde el ejercicio 4 en adelante. Priorice:
 - Algoritmos 1, 2, y 5 del ejercicio 4
 - Ejercicios 9, 10 y 13Puede valerse de los enlaces útiles de la página de la cátedra u otra referencia de programación en C.